IN THE CLAIMS:

1. (Currently amended) A multithreading processor, comprising:

a thread control unit;

a multithreaded register file having a plurality of registers; and

a plurality of hold latches, wherein

each of a plurality of the registers in the multithreaded register file and each of a plurality of the hold latches stores data representing a first instruction thread and a second instruction thread; and

the thread control unit provides a thread control ~~signal~~ signals to said hold latches and registers selecting a thread using said ~~data~~ thread control signals, wherein one of the thread control signals, applied to the hold latches, is delayed in time from another of the thread control signals, applied to the multithreaded register file, when selecting the thread to thereby accommodate pipelining of instructions by the multithreading processor.

2. (Currently amended) The multithreading processor as recited in claim 1, wherein the thread control unit via the thread control ~~signal~~ signals places at least one of the plurality of the hold latches and at least one of the plurality of the register files into an interleaving multithreading mode.

3. (Original) The multithreading processor as recited in claim 1, wherein the thread control unit, responsive to a determination that a latency in an instruction exceeding a first predetermined time has occurred in one of the two threads, sends control signals to said hold latches and register files for reading out data exclusively from the other of the two threads until a second predetermined time has elapsed.

4. (Currently amended) The multithreading processor as recited in claim 3, wherein the thread control unit ~~returns~~ places the plurality of hold latches and register files ~~to~~ into an interleaving multithreading mode after the expiration of the second time period.

5. (Currently amended) The multithreading processor as recited in claim 3, wherein the latency in an instruction exceeding a first predetermined time results from a load instruction that misses in a ~~datacache~~ data cache.

6. (Original) The multithreading processor as recited in claim 3, wherein the latency in an instruction exceeding a first predetermined time results from a mispredicted branch.

7. (Currently amended) A data processing system, comprising:

a memory unit;

a mixed-mode multithreading processor; and

a bus coupling the ~~multimedia~~ mixed-mode multithreading processor to the memory unit; wherein the ~~multimedia~~ mixed-mode multithreading processor comprises:

a thread control unit;

a multithreaded register file having a plurality of registers; and

a plurality of hold latches; wherein

each of a plurality of the registers in the multithreaded register file and each of a plurality of the hold latches stores data representing a first instruction thread and a second instruction thread; and

the thread control unit provides thread control signals to said hold latches and registers selecting a thread using said ~~data~~ thread control signals, wherein one of the thread control signals, applied to the hold latches, is delayed in time from another of the thread control signals, applied to the multithreaded register file, when selecting the thread to thereby accommodate pipelining of instructions by the mixed-mode multithreading processor.

8. (Currently amended) The data processing system as recited in claim 7, wherein the thread control unit via the thread control ~~signal~~ signals places at least one of the plurality of the hold latches and at least one of the plurality of the registers into an interleaving multithreading mode.

9. (Original) The data processing system as recited in claim 7, wherein the thread control unit, responsive to a determination that a latency in an instruction exceeding a first predetermined time has occurred in one of the two threads, sends control signals to said hold latches and registers for reading out data exclusively from the other of the two threads until a second predetermined time has elapsed.

10. (Original) The data processing system as recited in claim 9, wherein the thread control unit returns said hold latches and registers to an interleaving multithreading mode after the expiration of the second time period.

11. (Currently amended) The data processing system as recited in claim 9, wherein the latency in an instruction exceeding a first predetermined time results from a load instruction that misses in a ~~datacache~~ data cache.

12. (Original) The data processing system as recited in claim 9, wherein the latency in an instruction exceeding a first predetermined time results from a mispredicted branch.

13. (Currently amended) The data processing system as recited in claim 7, wherein the ~~multimedia~~ mixed-mode multithreading processor is a first ~~multimedia~~ mixed-mode multithreading processor, the thread control unit is a first thread control unit, the hold latches are first hold latches, the multithreaded register file is a first multithreaded register file, the plurality of registers are a plurality of first registers, and the data is a first data, and further comprising:

a second ~~multimedia~~ mixed-mode multithreading processor, wherein the second ~~multimedia~~ mixed-mode multithreading processor comprises:

a second thread control unit;

a second multithreaded register file having a plurality of second registers; and

a plurality of second hold latches, wherein

each of a plurality of the second registers and each of a plurality of the second hold latches stores second data representing a third instruction thread and a fourth instruction thread; and

the second thread control unit provides second thread control signals to said second hold latches and second registers selecting a second thread using said second ~~data~~ thread control signals, wherein one of the second thread control signals, applied to the second hold latches, is delayed in time from another of the second thread control signals, applied to the second multithreaded register file, when selecting the second thread to thereby accommodate pipelining of second instructions by the second mixed-mode multithreading processor.

14-19. (Canceled)

20. (Newly added) The multithreading processor as recited in claim 1, further comprising a plurality of flow through latches used to break multiple-cycle paths into distinct stages of a pipeline for pipelining the instructions.

21. (Newly added) The data processing system as recited in claim 7, further comprising a plurality of flow through latches used to break multiple-cycle paths into distinct stages of a pipeline for pipelining the instructions.